PROBLEM SOLVING WITH PROGRAMMING

(PSP)
COURSE PROJECT

Developed By:

Hrishitha.Sangineni
[2103A51244]

Under the Guidance of
Mr.RIYAZ MOHAMMED

Assistant Professor

Department Computer Science and Artificial Intelligence
SR University
Warangal.

PROBLEM STATEMENT:

Develop a C Application to store GROCERY LIST items in terms of (date, item_name , price ,code, quantity, brand ,manufacturing date). Store the data in a structure. Allocate memory for the structure by using DYNAMIC MEMORY MANAGEMENT FUNCTIONS.

Provide the functionality for below mentioned:

- Read 'n' item details dynamically.
- Sort (Ascending/Descending) 'n' items detailsaccording to:
  - day
  - Item_name
  - price
  - code
  - Quantity
  - manufacturing date
- Search 'n' passengers details according to :
  - day
  - item_name
  - price
  - code
  - qty
  - manufacturing date
- Print 'n' item's details

MODULES:

In this application all variables and structure are declared globally so that these variables and structure members can be accessed throughout the program at any function call. We can choose any function by using function calls which are declared in switch-case. In order to repeat the loop control statement (do-while) is used with condition. The memory allocation will be done in this program dynamically. The application asks the person who runs the program that how many passengers data he/she want to store.

In this application four modules are used.

- Read/Input

  In this module the application asks the person who runs the program to enter n items details.To give n items details for loop is used.

- Searching

  In this module searching of data is done according to the chosen wise.

  In this module there is a sub menu which asks to select the sorting wise by using if-else case. The sorting sub menu will be like to search specific groceries by their name code price and qty by different if blocks for given item_name code price and for qty is used and by this groceries items can be searced and details of it can be identified and found..

  In this module we used control statement (for) so that the application asks for multiple and more items

- Print

  In this module all the stored details of n items will be displayed on to the screen. In this module printf function and for loop are used.

KNOWLEDGE REQUIRED TO DEVELOP THIS APPLICATION

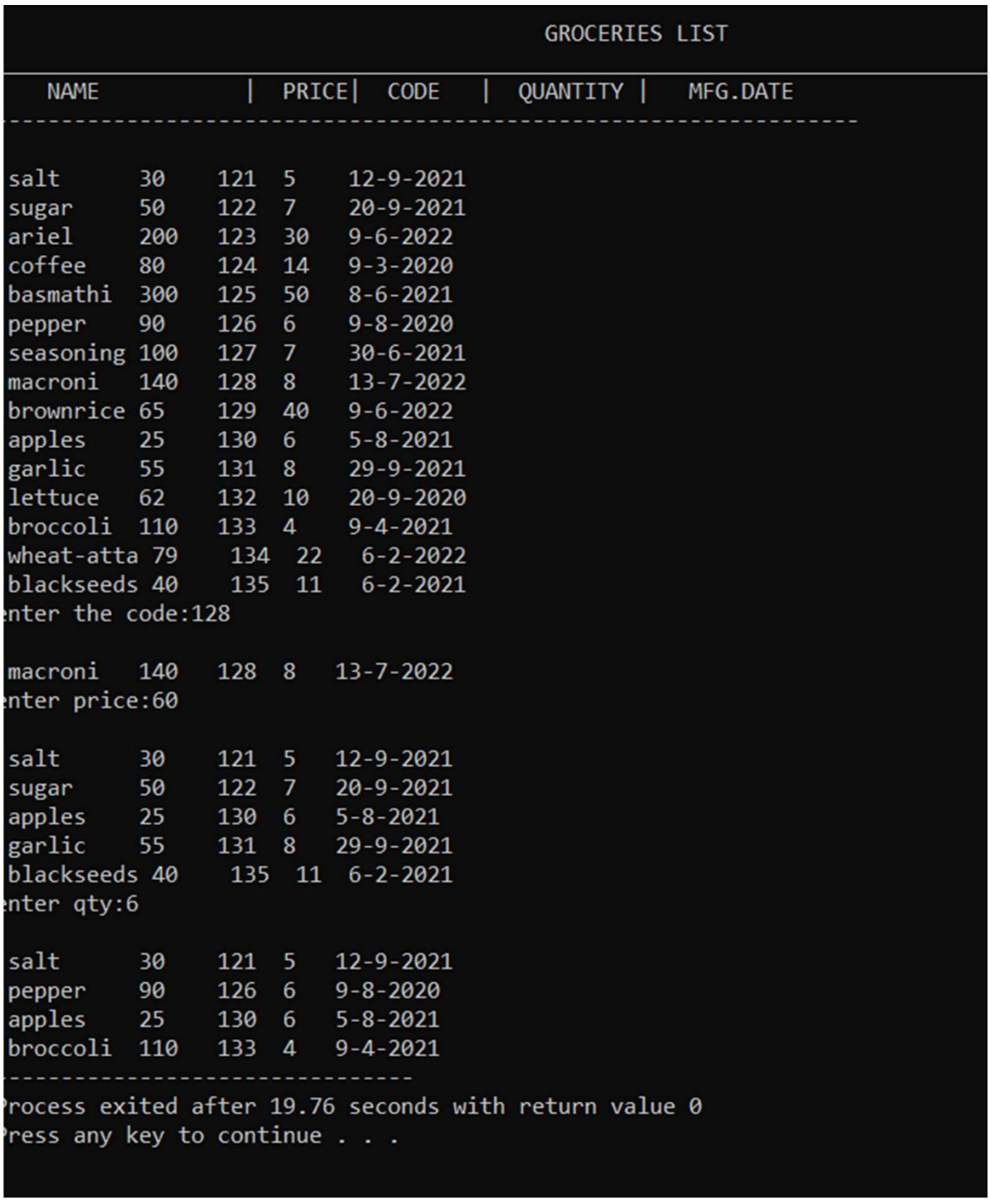
- Control Statements(if, if-else,)
- Loop Statements(for)
- Arrays (1-arrays)
- Structure (structures and nested structures)
- Dynamic Memory Allocation (malloc/ calloc/ realloc)

SOURCE CODE [HEADER FILE]:

```
#include<stdio.h>struct
date

{

  int  day; int
  month;int
  year;

};

struct details

{

        char item_name[20];int
        price;

        int code;int
        qty;

        struct date mfg;

};

void main()

{

        int      i,code,price,qty;
        struct details d[15];FILE
        *f3;

        /*f3=fopen("details.txt","w"); for(i=0;i<15;i++)

        {

        printf("\nenter item_name\tprice\tcode\tqty\tmanufacturing date(dd-mm-yyyy):");scanf("%s %d %d
        %d %d-%d-

%d",&d[i].item_name,&d[i].price,&d[i].code,&d[i].qty,&d[i].mfg.day,&d[i].mfg.month,&d[i]. mfg.year);

        }

fwrite(&d, sizeof(d),1,f3);
fclose(f3);*/
f3=fopen("details.txt","r");

fread(&d, sizeof(d),1,f3);

printf("\n                                GROCERIES LIST                                ");
printf("\n_____________________________________________________________________________

_____________________________________________");

printf("\n     NAME            | PRICE|  CODE  |  QUANTITY |   MFG.DATE \n");

printf("_____________________________________________________________________\n");
for(i=0;i<15;i++)

{
```

```
printf("\n %-9s %-5d %-4d %-3d %d-%d-
%d",d[i].item_name,d[i].price,d[i].code,d[i].qty,d[i].mfg.day,d[i].mfg.month,d[i].mfg.year);
}
printf("\nenter the code:");
scanf("%d",&code);
for(i=0;i<15;i++)
{

if(code==d[i].code)
{
 printf("\n %-9s %-5d %-4d %-3d %d-%d-
%d",d[i].item_name,d[i].price,d[i].code,d[i].qty,d[i].mfg.day,d[i].mfg.month,d[i].mfg.year);
}
}
printf("\nenter price:");
scanf("%d",&price);
for(i=0;i<15;i++)
{
        if(d[i].price<=price)
        {
        printf("\n %-9s %-5d %-4d %-3d %d-%d-
%d",d[i].item_name,d[i].price,d[i].code,d[i].qty,d[i].mfg.day,d[i].mfg.month,d[i].mfg.year);
        }
}
printf("\nenter qty:");
scanf("%d",&qty);
for(i=0;i<15;i++)
{
if(d[i].qty<=qty)
        {
printf("\n %-9s %-5d %-4d %-3d %d-%d-
%d",d[i].item_name,d[i].price,d[i].code,d[i].qty,d[i].mfg.day,d[i].mfg.month,d[i].mfg.year);
}
}
fclose(f3);
   }
```

OUTPUT:



```
                                        GROCERIES LIST
   NAME           |  PRICE|  CODE   |  QUANTITY |   MFG.DATE
-----------------------------------------------------------------------

salt       30    121  5    12-9-2021
sugar      50    122  7    20-9-2021
ariel      200   123  30   9-6-2022
coffee     80    124  14   9-3-2020
basmathi   300   125  50   8-6-2021
pepper     90    126  6    9-8-2020
seasoning  100   127  7    30-6-2021
macroni    140   128  8    13-7-2022
brownrice  65    129  40   9-6-2022
apples     25    130  6    5-8-2021
garlic     55    131  8    29-9-2021
lettuce    62    132  10   20-9-2020
broccoli   110   133  4    9-4-2021
wheat-atta 79     134  22   6-2-2022
blackseeds 40     135  11   6-2-2021
nter the code:128

macroni    140   128  8    13-7-2022
nter price:60

salt       30    121  5    12-9-2021
sugar      50    122  7    20-9-2021
apples     25    130  6    5-8-2021
garlic     55    131  8    29-9-2021
blackseeds 40     135  11   6-2-2021
nter qty:6

salt       30    121  5    12-9-2021
pepper     90    126  6    9-8-2020
apples     25    130  6    5-8-2021
broccoli   110   133  4    9-4-2021
--------------------------------
rocess exited after 19.76 seconds with return value 0
ress any key to continue . . .
```